

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-081455

(43)Date of publication of application : 28.03.1997

(51)Int.Cl.

G06F 12/08

G06F 13/00

(21)Application number : 07-232516

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 11.09.1995

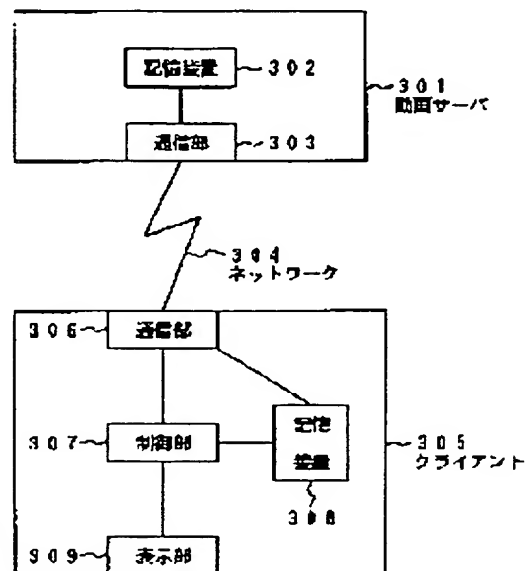
(72)Inventor : SUZUOKA SETSU

(54) CLIENT DEVICE SERVER DEVICE AND STORAGE DEVICE CONTROLLING METHOD

(57)Abstract:

PROBLEM TO BE SOLVED: To obtain desired continuous data without delay and interruption only if a few storage area is provided by holding (prefetch/caching) the leading part of continuous data.

SOLUTION: A moving picture server 301 is provided with a storage device 302 holding a moving picture data and a communication part 303 performing a communication control. Each client 305 is provided with a communication part 306 performing the communication via a network, a high speed storage device 308, a control part 307 and a display part 309. If the leading part of data is stored in a cache when data is read, the data of the leading part is processed after the reading request of the succeeding part is given, for instance. Because the leading part is only the quantity corresponding to data reading delay time and the data of the succeeding part arrives before the data is processed, the succeeding processing can be performed without interruption by using the data.



LEGAL STATUS

[Date of request for examination] 11.09.2000

[Date of sending the examiner's decision of rejection] 01.06.2004

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11) 特許出願公開番号

特開平9-81455

(43) 公開日 平成9年(1997)3月28日

(51) IntCl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 12/08		7623-5B	G 0 6 F 12/08	D
13/00	3 5 7		13/00	3 5 7 Z

審査請求 未請求 請求項の数 5 O L (全 15 頁)

(21) 出願番号 特願平7-232516

(22) 出願日 平成7年(1995)9月11日

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72) 発明者 鈴木 節

神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内

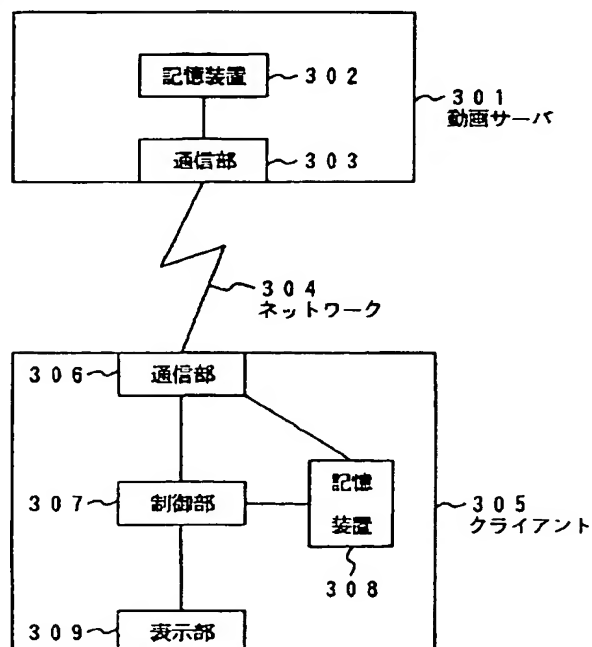
(74) 代理人 弁理士 鈴木 武彦

(54) 【発明の名称】 クライアント装置、サーバ装置及び記憶装置制御方法

(57) 【要約】

【課題】 少ない記憶領域を備えただけで、遅延なくかつ途切れなしに所望の連続データを得ることができるクライアント装置を提供する。

【解決手段】 連続して処理する一纏まりの連続データを記憶した記憶装置から得た連続データをデータ処理するクライアント装置において、前記記憶装置に記憶された前記連続データのうち、前記データ処理に要する時間が前記記憶装置の読み出し遅延に相当する量またはそれを越える量のデータを先頭から連続して保持しておくデータ先頭部保持手段と、前記データ先頭部保持手段に保持されている前記連続データの先頭から前記データ処理が開始されるのと同時にまたは相前後して、前記記憶装置に記憶された前記連続データのうち前記データ先頭部保持手段に保持されているデータの続き位置またはそれより所定データ量前の位置から後続するデータに対する読み出し要求を出す手段とを備える。



【特許請求の範囲】

【請求項 1】連続して処理する一纏まりの連続データを記憶した記憶装置から得た連続データをデータ処理するクライアント装置において、

前記記憶装置に記憶された前記連続データのうち、前記データ処理に要する時間が前記記憶装置の読み出し遅延に相当する量またはそれを越える量のデータを先頭から連続して保持しておくデータ先頭部保持手段と、

前記データ先頭部保持手段に保持されている前記連続データの先頭から前記データ処理が開始されるのと同時にまたは相前後して、前記記憶装置に記憶された前記連続データのうち前記データ先頭部保持手段に保持されているデータの続き位置またはそれより所定データ量前の位置から後続するデータに対する読み出し要求を出す手段とを備えたことを特徴とするクライアント装置。

【請求項 2】前記記憶装置に複数記憶された前記連続データのうち前記データ先頭部保持手段に先頭部を保持しておく候補と各候補の優先度からなる情報に基づいて、前記データ先頭部保持手段に先頭部を保持しておく連続データを決定する手段をさらに備えたことを特徴とする請求項 1 に記載のクライアント装置。

【請求項 3】ネットワークにより請求項 1 に記載のクライアント装置が接続され、請求項 1 に記載の記憶装置を備え、該クライアント装置からの要求に応じて該記憶装置に記憶されたデータを提供するサーバ装置であって、前記記憶装置へのアクセスの集中度および前記ネットワークの混雑度に応じて、読み出し要求の種類ごとに許可または禁止を指示するための読み出し条件を決定する手段と、

決定された前記読み出し条件を前記クライアント装置に指示する手段とを備え、

前記読み出し条件は、読み出しの余裕度が高い方から順に、任意の読み出し許可、データ先頭部プリフェッチ読み出し禁止、データ先頭部プリフェッチ読み出しおよび前記データ先頭部保持手段に保持しているデータ先頭部を用いた後続部読み出し禁止とすることを特徴とするサーバ装置。

【請求項 4】所定のデータ処理を行なうクライアント装置が、連続して処理する一纏まりの連続データを記憶した記憶装置から処理対象データを得るための記憶装置制御方法において、

前記記憶装置に記憶された前記連続データのうち所定のものについて、前記データ処理に要する時間が前記記憶装置の読み出し遅延に相当する量またはそれを越える量のデータを該連続データの先頭から連続して保持しておく、

先頭部が保持されている前記連続データに対する処理要求に応じ、保持されている前記連続データの先頭から前記データ処理を開始し、

前記データ処理を開始するのと同時にまたは相前後し

て、前記記憶装置に記憶された前記連続データのうち前記データ先頭部保持手段に保持されているデータの続き位置またはそれより所定データ量前の位置から後続するデータについての読み出し要求を出すことを特徴とする記憶装置制御方法。

【請求項 5】連続して処理する一纏まりの連続データを記憶した記憶装置を有するサーバ装置と、ネットワークを介して該サーバ装置から所望の連続データを得るとともに、所定の連続データについてデータ先頭部プリフェッチ読み出しおよび後続部読み出しを行なうクライアント装置とを備えたシステムにおいて該サーバ装置が該クライアント装置からの読み出しアクセスを制御するための記憶装置制御方法であって、

前記記憶装置へのアクセスの集中度および前記ネットワークの混雑度に応じて、読み出し条件を、任意の読み出し許可、データ先頭部プリフェッチ読み出し禁止、またはデータ先頭部プリフェッチ読み出しおよび前記データ先頭部保持手段に保持しているデータ先頭部を用いた後続部読み出し禁止のうちから決定し、

決定された前記読み出し条件を前記クライアント装置に指示することを特徴とする記憶装置制御方法。

【発明の詳細な説明】

【0001】

【発明の属する利用分野】本発明は、読み出し遅延の大きい記憶装置からデータを遅延なくかつ途切れなしに読み出すことを可能とするクライアント装置、サーバ装置及び記憶装置制御方法に関する。

【0002】

【従来の技術】遅い記憶装置から高速にデータを読み出すかのようにみせかける技術として、キャッシング (caching) とプリフェッチ (prefetch) とがある。

【0003】これらの技術を説明するのに先だって、記憶装置から処理装置にデータを読み込んで処理するという最も基本的な構成・動作を図 12 を用いて説明する。図 12 に示すように、データ処理装置 101 と記憶装置 102 がデータ伝送路 103 によりつながれているものとする。ここで、記憶装置 101 は半導体メモリ、ディスク装置、光ディスク (CD-ROM を含む) などであり、データ伝送路 103 はバス、ケーブル、ネットワークなどである。このような構成において、データ処理装置 101 は、必要なデータを記憶装置 102 に要求し、データ伝送路 103 を介して要求したデータを受取り、それを処理する。

【0004】ここで、図 12 の各部の処理速度を以下のように記号で表す。データ処理装置 101 がデータを処理する速度を a (bytes/sec) とし、データ伝送路 103 の転送速度を b (bytes/sec) とし、記憶装置 102 において、データを読み出してデータ伝送路 103 に送り出す速度を c (bytes/sec)

c)とする。一般に、 $b < a$ もしくは $c < a$ であり、データ処理装置101がデータ待ち (starvation) を起こし、性能が低下する傾向がある。

【0005】この問題を軽減する方法としてキャッシング技術が考案されている(ヘネシー&パターソン著、「コンピュータ・アーキテクチャ」、日経BP、1992年)。キャッシングとは、一度読んだことのあるデータを身近の高速メモリ(キャッシュ; cacheと呼ぶ)に格納しておき、次に同じデータを読み出す必要が生じた場合には、オリジナルのデータがある場所からではなく、キャッシュからデータを読み出す技術である。これにより、2度目から高速にデータを読み出すことを可能にする。

【0006】上記のようなキャッシュを備えたシステムの構成を図13に示す。ここで、データ処理装置201、記憶装置202、データ伝送路203は図12と同じものである。新たに加わったのは、高速メモリ(キャッシュ)204、および高速メモリとデータ処理装置を結ぶ高速データ伝送路205である。ここで、記憶装置202から読み出されたデータは、データ伝送路203を介して伝わり、高速メモリ204にデータのコピーを作った上で、データ処理装置201に伝わる。二度目に同じデータにアクセスした場合には、高速メモリ202上のデータが高速データ伝送路205を介してデータ処理装置201に伝わる。ここで、高速メモリのアクセス速度を d (bytes/sec)とし、高速データ伝送路の伝送速度を e (bytes/sec)とする。 $d \geq a$ かつ $e \geq a$ であるように作っておくならば、データ処理装置201に滞りなくデータが供給される。

【0007】しかしながら、キャッシング技術でデータに素早くアクセスすることができるのは、同一データについての二度目のアクセスからであり、最初のデータアクセスは依然として遅いことになる。

【0008】そこで、最初から高速にアクセスすることができるよう、プリフェッチという技術が考案されている。この技術では、将来読み出す可能性が高いデータがあらかじめ分かっている場合に、そのデータを事前に読み出しておく。したがって、この方法は予測が確からしい場合にのみ効力を発揮する。

【0009】従来、データのプリフェッチやキャッシュは、数値データに対して行われていた。この場合、データはデータ処理装置201内の例えばベクトルプロセッサのような高速演算器で処理がなされる。 $b < a$ もしくは $c < a$ であるが、高速メモリ204と高速データ伝送路205を用いて $d \geq a$ かつ $e \geq a$ を達成し、データ読み出しのボトルネックを解消している。

【0010】ここで、注意すべき点は記憶装置202からのデータの読み出しは基本的に陥路になっているの

で、データ処理装置201がよどみなくデータを処理できるためには、データ処理装置201で使うデータが全て高速メモリ204に入っていないなければならない。しかし、この条件を満たすためには巨大な高速メモリが必要となり現実的ではない。実製品では、この高速メモリの容量は小さく、記憶装置202の必要なデータを全て高速メモリ204に格納しておくことはできない。

【0011】プリフェッチの別の用法としては、プロセッサの命令のプリフェッチという手法がある(ヘネシー&パターソン著、「コンピュータ・アーキテクチャ」、日経BP、1992年)。

【0012】これは、これから必要になると予測される命令を数語程度をあらかじめプロセッサ内の命令キャッシュに格納しておく手法であり、16-bit以上の多くのマイクロプロセッサでは備えている技術である。この場合には、 $b \geq a$ かつ $c \geq a$ であり、命令のアクセス遅延が問題である。このプリフェッチが数語でも効果があるのは、 b 、 c に比べて a が充分遅いからである。また、予測がはずれてプリフェッチバッファが空になり命令が途切れなく供給できなくとも、性能が一時低下するだけでそれ以上悪いことが起こらないからである。すなわち、この場合にはデータの連続供給性は保証されていない。

【0013】ところで、記憶装置に記憶した連続データをサーバからクライアントに供給するシステム、例えば動画を動画サーバからクライアントに供給するシステムでは、データ転送はネットワークにATMなどを用いて充分高速に実現できるが、記憶装置の読み出し遅延のために動画データを最初にアクセスする頭出し時間を必要とする。したがって、従来は、映像が出はじめた後は連続的に再生できるが、動画を要求してから映像が出はじめるまでが遅いという欠点があった。

【0014】もちろん、全てのデータをキャッシングしておけば、要求とともに映像の再生を開始し、かつ、途切れることなく高速アクセスすることが可能であるが、動画データをキャッシングするためには莫大なキャッシング容量が必要であり、非現実的である。また、データを全てプリフェッチするのは膨大な時間がかかり、やはり現実的でない。

【0015】また、ネットワークを用いないCD-ROMなどの場合でも、ヘッドの位置によって回転速度が異なるので、CD-ROMのデータをランダムにアクセスすると、回転速度を調整するために頭出し時間が必要である。

【0016】さらに、複数枚のCD-ROMをCD-ROMイクスチェンジャを用いてロボットアームが望ましいCD-ROMをCD-ROMドライブに運んでCD-ROMからデータを読み出す場合には、ロボットアームが動作する時間(2~3秒)の読み出し遅延がある。マルチメディア時代に向い、様々なデータを切替えて見る

操作が多用される傾向にあるので、切替え時の頭だし時間は極めて大きな問題となる。

【0017】

【発明が解決しようとする課題】従来は、サーバの管理する連続データをクライアントに供給する場合、記憶装置におけるデータの読み出し遅延（頭出し遅延）が律速となり、クライアントが要求を出してからデータの先頭を得るまでに一定の時間を要した。

【0018】もちろん、全てのデータをキャッシングしておけば、上記問題は解決するが、莫大なキャッシング容量を必要とし非現実的である。また、データを全てプリフェッチするには膨大な時間がかかる。

【0019】また、CD-ROMなどの二次記憶装置を備えたシステムでも、回転速度調整のための頭出し時間や情報記憶媒体自体の交換時間等によりアクセス遅延が生じる問題点があった。

【0020】本発明は、上記事情を考慮してなされたものであり、少ない記憶領域を備えただけで、遅延なくかつ途切れなしに所望の連続データを得ることができるクライアント装置、サーバ装置及び記憶装置制御方法を提供することを目的とする。

【0021】

【課題を解決するための手段】本発明（請求項1）は、連続して処理する一纏まりの連続データを記憶した記憶装置から得た連続データをデータ処理するクライアント装置において、前記記憶装置に記憶された前記連続データのうち、前記データ処理に要する時間が前記記憶装置の読み出し遅延に相当する量またはそれを越える量のデータを先頭から連続して保持しておくデータ先頭部保持手段と、前記データ先頭部保持手段に保持されている前記連続データの先頭から前記データ処理が開始されるのと同時にまたは相前後して、前記記憶装置に記憶された前記連続データのうち前記データ先頭部保持手段に保持されているデータの続き位置またはそれより所定データ量前の位置から後続するデータに対する読み出し要求を出す手段とを備えたことを特徴とする。

【0022】本発明（請求項2）は、前記記憶装置に複数記憶された前記連続データのうち前記データ先頭部保持手段に先頭部を保持しておく候補と各候補の優先度からなる情報に基づいて、前記データ先頭部保持手段に先頭部を保持しておく連続データを決定する手段をさらに備えたことを特徴とする請求項1に記載のクライアント装置。

【0023】好ましくは、前記情報における前記優先度を、前記連続データの実際の読み出し状況に応じて更新する手段をさらに備えたことを特徴とする。これらの発明は1台の記憶装置に対してアクセス可能なクライアント装置が1台でも複数台でも適用可能である。例えば、記憶装置がネットワークにつながって複数のクライアント装置が存在する場合でも、記憶装置がローカルに接続

されたCD-ROMであっても適用可能である。

【0024】また、次の発明は、クライアント装置が複数存在する場合または複数のクライアント装置が共有のネットワークを利用する場合に適用可能な発明である。本発明（請求項3）は、ネットワークにより請求項1に記載のクライアント装置が接続され、請求項1に記載の記憶装置を備え、該クライアント装置からの要求に応じて該記憶装置に記憶されたデータを提供するサーバ装置であって、前記記憶装置へのアクセスの集中度および前記ネットワークの混雑度に応じて、読み出し要求の種類ごとに許可または禁止を指示するための読み出し条件を決定する手段と、決定された前記読み出し条件を前記クライアント装置に指示する手段とを備え、前記読み出し条件は、読み出しの余裕度が高い方から順に、任意の読み出し許可、データ先頭部プリフェッチ読み出し禁止、データ先頭部プリフェッチ読み出しおよび前記データ先頭部保持手段に保持しているデータ先頭部を用いた後続部読み出し禁止とすることを特徴とする。

【0025】本発明（請求項4）は、所定のデータ処理を行なうクライアント装置が、連続して処理する一纏まりの連続データを記憶した記憶装置から処理対象データを得るための記憶装置制御方法において、前記記憶装置に記憶された前記連続データのうち所定のものについて、前記データ処理に要する時間が前記記憶装置の読み出し遅延に相当する量またはそれを越える量のデータを該連続データの先頭から連続して保持しておき、先頭部が保持されている前記連続データに対する処理要求に応じ、保持されている前記連続データの先頭から前記データ処理を開始し、前記データ処理を開始すると同時にまたは相前後して、前記記憶装置に記憶された前記連続データのうち前記データ先頭部保持手段に保持されているデータの続き位置またはそれより所定データ量前の位置から後続するデータについての読み出し要求を出すことを特徴とする。

【0026】本発明（請求項5）は、連続して処理する一纏まりの連続データを記憶した記憶装置を有するサーバ装置と、ネットワークを介して該サーバ装置から所望の連続データを得るとともに、所定の連続データについてデータ先頭部プリフェッチ読み出しおよび後続部読み出しを行なうクライアント装置とを備えたシステムにおいて該サーバ装置が該クライアント装置からの読み出しアクセスを制御するための記憶装置制御方法であって、前記記憶装置へのアクセスの集中度および前記ネットワークの混雑度に応じて、読み出し条件を、任意の読み出し許可、データ先頭部プリフェッチ読み出し禁止、またはデータ先頭部プリフェッチ読み出しおよび前記データ先頭部保持手段に保持しているデータ先頭部を用いた後続部読み出し禁止のうちから決定し、決定された前記読み出し条件を前記クライアント装置に指示することを特徴とする。

【0027】好ましくは、前記記憶装置へのアクセスが集中していると判断された場合には、少なくともデータ先頭部プリフェッチ読み出し禁止を含む読み出し条件を決定し、クライアント装置はデータ先頭部プリフェッチをせず、サーバ装置はデータ先頭部プリフェッチ要求をキャンセルすることを特徴とする。

【0028】（作用）本発明（請求項1、4）では、データの転送速度は処理速度に匹敵するだけの容量を持つが、読み出しの遅延（latency）の大きな記憶装置からのデータの読み出しを行う場合に、処理に必要なデータの全てをキャッシングしたり、プリフェッチするのではなく、データの先頭部分の読み出しの遅延時間に相当する量だけをキャッシングしたり、プリフェッチを行なう。

【0029】データを読み出す場合には、データの先頭部分がキャッシュに格納されているならば、例えばその後続部分の読み出し要求を出した後にその先頭部分のデータを処理する。先頭部分はデータ読み出し遅延時間に相当する分量だけあり、そのデータが処理終わる前には後続部のデータが到着するので、そのデータを用いて切れ目なく続きの処理を行なうことができる。

【0030】このように、少ないデータのキャッシングやプリフェッチングにより、連続データを遅延時間少なく処理することができる。本発明（請求項2）によれば、プリフェッチすべきデータの情報が優先度付きで与えられるので、効率のよいプリフェッチが可能となる。

【0031】また、優先度を実際の読み出し状況に応じて更新するにすれば、プリフェッチすべきデータの情報が今までに一連のデータをアクセスしてきた過去のデータから学習されるので、より現実的に即したプリフェッチ情報となり有効である。

【0032】本発明（請求項3、5）では、複数のクライアント装置が一台のサーバ装置へアクセスしようとして、記憶装置内の制御やサーバ装置へのネットワークが混雑したような場合には、緊急性の低いデータアクセスを禁止するために、プリフェッチ要求は出さないようにクライアント装置に指示する。なぜならば、プリフェッチで要求されたデータは必ずしも使われないので重要度が低いからである。

【0033】また、一台の記憶装置にアクセスが集中して、所定の遅延時間内にデータを提供できないような場合には、クライアント装置がキャッシュ内の先頭データを用いて、後続データを要求されないようにする。もし、非常に混雑している時にそのような処理を行なうと、先頭部分の処理が終了しても、後続のデータが要求元のクライアント装置に到着しないことがあるからである。

【0034】このように、データが途切れる可能性がある場合には、後続部の読み出しを禁止するようにサーバ装置からクライアント装置に通達する。後続部の読み出

しを禁止するのは、頭出しが速くても、その後で先頭部分の処理が終了したところでデータが途切れるよりは、頭出しが遅くても途切れないデータ供給を行なうためである。

【0035】また、例えばサーバ装置側でデータの要求に対応しきれないと判断し、プリフェッチをしないようにクライアント装置に指示を出しても、行き違いでそのような指示を出した後にプリフェッチ要求が来ることもある。したがって、このような場合には、前述のようにそのプリフェッチ要求をキャンセルするのが好ましく、また、クライアント装置側でも、読み出し遅延が予想していた時間よりも長い場合には自主的にプリフェッチ要求を出すことを控えるのが好ましい。

【0036】

【発明の実施の形態】以下、図面を参照しながら発明の実施の形態を説明する。まず、以下で用いる語句を定義する。「プリフェッチ」とは、データを実際に使用する状況が生じる前に、あらかじめ読み出ししておくことを言う。プリフェッチは、データアクセス時の遅延を隠蔽する効果を有する。

【0037】「キャッシング」とは、一度読み出したデータをローカルなより高速のメモリに格納しておき、次にそのデータが必要な場合には、元々そのデータがあったところからではなく、該高速メモリから読み出すことを言う。

【0038】図1に、この実施の形態に係るシステム構成図を示す。図1に示すように、動画サーバ301と複数のクライアント305とがネットワーク304を介して接続されている。なお、図1では、1台のクライアントのみ示し、他のクライアントは省略した。

【0039】動画サーバ301は、動画データを保持する記憶装置302と通信制御を行う通信部303を備えている。一方、各クライアント305は、ネットワークを介した通信を行う通信部306、高速記憶装置308、制御部307および表示部309を備えている。

【0040】図1のシステムにおけるデータの流れは次のようになる。図示しない入力装置からユーザにより再生要求が入力されるなどすると、これに応じて制御部307で動画再生要求が発生する。この動画再生要求は、通信部306を通り、ネットワーク304を介して動画サーバ301の通信部303に到達する。これが記憶装置302に伝わり、要求されたデータが通信部303、ネットワーク304、通信部306を通して、高速記憶装置308に格納される。このデータを制御部307で読みとり、コード変換やタイミング処理を行って表示部309に表示する。

【0041】次に、動画サーバ301側での処理について説明する。まず、記憶装置302内の制御装置（図示せず）は、アクセスの集中が起こっているか否かを判断するとともに、ネットワークの混雑の度合いを調べてい

る。

【0042】もしプリフェッチ要求が多く、それがアクセス集中を招いていると判断した場合には、そのことをネットワーク309を介してクライアント305に伝える。また、所定の頭出し待ち時間を守れないと判断した場合にも、その旨をクライアント305に知らせる。また、アクセスの集中が緩和された場合には、その制限を取り除くことをクライアント305に通達する。さらに、プリフェッチを禁止したのちにきたプリフェッチの要求に対しては、要求を却下する返事を要求を出したクライアント305に返す。なお、プリフェッチの情報として、次に（もしくは近い将来に）アクセスする可能性の高いと思われる動画データのリストを優先度付きでクライアント305に伝える。

【0043】次に、クライアント305側の処理について説明する。図2にクライアント305側に持つデータ構造を示す。このデータ構造は、キャッシングもしくはプリフェッチによって格納された先頭部データを管理するために用いる。

【0044】動画データは、識別番号idで区別される。識別番号idは非負整数である。この識別番号idをもとにハッシュ表を用いて、先頭部データを管理する。これには、配列hash、配列id、配列nextを用いる。これらの配列の初期値は全て-1である。配列hashはH個の要素を持ち、配列idと配列nextはそれぞれN個の要素を持つ。

【0045】さらに、キャッシュのアクセス頻度cache、プリフェッチの優先度pref、具体的なデータの居場所を表すadrを用いる。これらは全てN個の要素を持つ配列である。prefの値が大きい方を優先度

【0046】具体的なデータの存在場所は半導体メモリであっても、磁気ディスクであってもよい。この値は初期状態から設定されている。すなわち、メモリ上の領域を取った上でadrが設定されていなければならない。

【0047】また、本実施形態では、上記のものの他に以下の変数を用いる。

「busy_status」

busy_statusは、動画サーバから与えられる混雑の度合いに応じた状態を表す。busy_statusのとり値と、各値により示される状態は以下のとおりである。

0... 通常

1... プリフェッチ禁止

2... プリフェッチ禁止及びローカルに格納しているデータ先頭部を用いた後続部読み出し禁止

「net_using」

net_usingは、ネットワークを介して動画サーバにアクセス中か否かを示す状態変数である。net_usingのとり値と、各値により示される状態は以下

のとおりである。

0... 非アクセス中（初期状態）

1... アクセス中

「head」headは、データの先頭部を格納すべき場所adr[head]を示す識別子である。headは、格納する必要がない場合には-1となっており、-1が初期状態である。

【0048】「reading」readingは、プリフェッチ要求ではなく実際に動画データを読み込もうとしているかどうかを示す状態変数である。readingのとり値と、各値により示される状態は以下のとおりである。

0... 読み込もうとしていない（初期状態）

1... 読み込もうとしている

「pref_n」pref_nは、動画サーバから送られてくるプリフェッチ情報のアイテム数である。

【0049】「pref_id」pref_idは、動画サーバから送られてくるプリフェッチすべきデータの識別子の配列であり、その要素数はpref_nである。

【0050】「pref_pr」pref_prは、動画サーバから送られてくるプリフェッチすべきデータの優先度の配列である。pref_prは、pref_idに対応しており、その要素数はpref_nである。このpref_prが高い順に並べられて送られてくる。この各要素の値の最大値はCである。

【0051】「ipr」iprは、次に処理すべきpref_idを示す識別子であり、次にpref_id[ipr]を処理すべきであるということを示す。

【0052】「C」Cは、最大の優先度を表す正定数である。次に、クライアント305に係わるフローチャートを説明する。

【0053】ここで、以下の説明で参照するフローチャートを列挙する。図3に番号iiの動画データを読み出す手順read(ii)を示す。図4に時計割り込み時の処理を示す。図5にネットワークからのメッセージの到着時の処理を示す。図6に識別子iiの動画データの先頭データが格納されている位置を調べるfind(ii)を示す。図7に識別子iiの動画データの先頭データをj番目の位置に格納するための処理add(ii, j)を示す。図8に識別子iiの動画データの先頭データを削除するための処理delete(ii)を示す。図9に新たな動画データの先頭データを格納すべき位置を検索する関数find_placeを示す。図10に各データの優先度を落とすdecrement処理を示す。図11に保持しているデータのうちの最も優先度の低いものを探すmin_prという処理を示す。

【0054】まず、図3を参照しながら番号iiの動画データを読み出す手順read(ii)について説明する。まず、変数readingを1に設定する（ステッ

プS501)。この設定はread(ii)の最後で0に戻すまで変わらない(ステップS515)。次に、自クライアントからデータの要求があればそれが終わるまで待つ(ステップS502)。ネットワークからデータが届くと割り込み処理によって処理されるので、ステップS502は無限ループになることはない。次に、find(ii)で識別子iiのデータの先頭部が既に格納されているかどうかを調べる(ステップS503)。そして、ネットワークを使用中にする(ステップS504)。

【0055】もし、ローカルに格納しているデータ先頭部を用いた後続部読み出しが許可されていてかつ先頭部が格納されているならば(ステップS505)、その続きからの読み出し要求を出し(ステップS506)、格納されているデータの優先度を下げ(ステップS507)、今回アクセスがあったj番目の要素については優先度を上げ(ステップS508)、格納されている先頭部のデータadr[j]を処理するように指示を出す(ステップS509)。

【0056】一方、ローカルに格納しているデータ先頭部を用いた後続部読み出しが禁止されているかもしくは先頭部が格納されていないならば(ステップS505)、最初からの読み出しを要求する(ステップS510)。もし先頭部のデータが格納されていないならば(ステップS511)、その先頭部を格納する場所を探し、その場所をheadとする(ステップS512)。先頭部データの優先度を一律に下げた後に(ステップS513)、今回アクセスがあるj番目の要素については優先度を上げる(ステップS514)。

【0057】上記のフローでは簡単のため、ローカルに格納しているデータ先頭部を用いた後続部読み出しが禁止されているが、先頭部が格納されている場合でも、先頭部からデータを読み込むようになっている。しかし、より一層の高速化のためには、そのような場合には、後続部分から読み出すように制御することもできる。

【0058】次に、図4を参照しながら時計割り込み時の処理について説明する。時計割り込みが入ってきたときにプリフェッチを行うが、プリフェッチは優先度の低い処理であるので、処理を目的として動画データを読みとり中であつたり(ステップS601)、何らかの目的(別のプリフェッチである場合も含む)でデータを読みとり中であつたり(ステップS602)、動画サーバからプリフェッチが禁止されている場合(ステップS603)には、プリフェッチをせずに本処理を終了させる。

【0059】サーバから送られてきたプリフェッチのヒントはn_pref個であるので、現在処理中のものiprがこれを越えていれば(ステップS604)、プリフェッチするものがこれ以上ないので本処理を終了する。

【0060】ステップS604～S607では、プリフェッチすべきデータを探す。既にローカルに格納されている場合はスキップする(ステップS605～S607)。もしあれば、そのデータを格納すべき位置を検索する(ステップS608)。その検索された位置であるjと優先度を比べ(ステップS609)、もしiprの方が低ければ、iprを最大の整数MAXINTにし(ステップS610)、これ以上プリフェッチをしないようにする。なぜならば、pref_idは優先度の高い順に並んでいるからである。現在格納されているデータの中で一番低い優先度よりもpref_id[ipr]の優先度は低いので、それ以上先を先を調べる必要はない。

【0061】もし優先度が高いならば(ステップS609)、j番目の要素を削除しそこに今回のプリフェッチの先頭部データを格納するように設定する(ステップS611)。pref_id[ipr]番のデータの先頭部分をプリフェッチするように指示を出す(ステップS612)。最後にiprをインクリメントする(ステップS613)。

【0062】次に、図5を参照しながらネットワークからのメッセージの到着時の処理について説明する。サーバからメッセージが到着すると割り込みが入り、この処理が実行される。まず、メッセージの種類を判別する(ステップS701)。

【0063】もし、それが要求したデータ(ここでは動画データ)であるならば、その先頭部をキャッシングするかどうかを判断する(ステップS702)。もし、キャッシングするならば、受信メッセージの先頭から所定の大きさの部分をadr[head]で指定された領域に格納し(ステップS703)、headを-1にして無効にする(ステップS704)。なお、プリフェッチの要求時もステップS703、S704の処理を実行する。

【0064】これがプリフェッチに対する要求の結果得られたデータでないならば(ステップS705)、そのデータを処理する(ステップS706)。例えば、そのデータが動画である場合には再生する。そして、net_usingに0を代入してnet_usingを非アクセス中(初期状態)にする(ステップS707)。

【0065】もしメッセージの種類がシステムメッセージであるならば(ステップS701)、メッセージの内容により個々別々の処理を行う。まず、プリフェッチ失敗のメッセージであるならば(ステップS708)、用意しておいたプリフェッチ部の格納領域headを解放し(ステップS709)、headに-1を代入してheadを無効にする(ステップS710)。そして、net_usingに0を代入してnet_usingを非アクセス中(初期状態)にする(ステップS707)。

【0066】また、`busy_status`の変更のメッセージであるならば（ステップS711）、指定されたように`busy_satus`を変更する（ステップS712）。何をプリフェッチすべきかに関する情報であるならば（ステップS713）、その情報に従って、`pref_n`、`pref_id`、`pref_pr`を設定し（ステップS714）、変数`ipr`を0に初期化する（ステップS715）。

【0067】その他のシステムメッセージの処理はステップS716で行う。次に、図6を参照しながら識別子`ii`の動画データの先頭データが格納されている位置を調べる手順`find(ii)`について説明する。

【0068】検索にはハッシュ法を用いている。識別番号`ii`に対応する動画データの先頭部分のデータのある場所はハッシュ表`hash`によって管理されている。この配列`hash`はH個の要素を持つ。

【0069】ハッシュ関数は簡単のためHで割った剰余を用いる（ステップS801）。`ii`番のデータがあるかどうかについては`hash[ii]`を調べる。これが-1ならばそのデータはないことを意味する（ステップS802）。この場合は、なかったという意味で-1を返す（ステップS808）。もしあれば、`hash[i]`の値を変数`j`に設定する（ステップS803）。`id[j]`が`ii`であるようなものがあるかどうかを`next`をたどりながら検索する（ステップS804～S806）、もしあれば（ステップS804）、その`j`を返し（ステップS807）、なければ-1を返す（ステップS808）。

【0070】次に、図7を参照しながら識別子`ii`の動画データの先頭データを`j`番目の位置に格納するための処理`add(ii, j)`について説明する。まず、ハッシュ関数を引き（ステップS901）、そのハッシュ値`i`を持つ物がすでにあるかどうかを調べる（ステップS902）。もしあれば、新たに加える要素`j`の`next`にいままでつながっていたものをつなげ（ステップS903）、なければ終りの意味で`next`を-1に設定する（ステップS904）。そして`j`に関して配列の要素を初期化する（ステップS905）。

【0071】次に、図8を参照しながら識別子`ii`の動画データの先頭データを削除するための処理`delete(ii)`について説明する。まず、ハッシュ関数を引き（ステップS1001）、削除すべきものがあるべき場所を探す（ステップS1002～S1004）。`id[j]`が`ii`に一致するまで検索を行う（ステップS1003）。ここで変数`k`は変数`j`の一つ前の要素を指している。これによりステップS1005～S1007にあるように`ii`を削除した後にも正しく他の要素が連結されているようにする。

【0072】次に、新たな動画データの先頭データを格納すべき位置を検索する処理について説明する。図9に

新たな動画データの先頭データを格納すべき位置を検索する関数`find_place`を示す。

【0073】場所としては、総合優先度が負であるものが選ばれる。それが無い場合には、総合優先度が正で最少のものが選ばれる。各要素は優先度として`cache`と`pref`を持つが、総合優先度として両者の和を用いる。

【0074】変数`j`は要素を一つ一つ辿るための識別子であり、0から始まり（ステップS1101）、インクリメントしながら（ステップS1107）、Nまで走査する（ステップS1108）。最少の優先度`min`の初期値は2Cとする（ステップS1101）。なお、優先度はこの値を超えないようになっている。

【0075】走査中に`id[j]`が負であるものが見つければ（ステップS1102）、その領域は未使用なので、使うことにし、`j`を返す（ステップS1111）。各走査において、`id[j]`が負でなければ（ステップS1102）、総合優先度`v`を求めるが（ステップS1103）、もしそれが負であれば（ステップS1104）、その領域を用いる。この領域は既に使用されているので、領域を解放し（ステップS1112、S1109）、その番号を返す（ステップS1110）。

【0076】また、`id[j]`が負でなく（ステップS1102）、総合優先度`v`が負でない場合（ステップS1104）、総合優先度`v`が今まで保持しているものよりも小さいければ、総合優先度`v`で最少の優先度`min`を置き換えを行ない（ステップS1105、S1106）、総合優先度が正で最少のものを選ぶ。

【0077】次に、各データの優先度を落とす`decrement`処理を示す。図10に`decrement`処理を示す。この処理ではキャッシングされている各要素の優先度を下げる。ただし、優先度が既に負であるものに対しては作用させない（ステップS1202、S1204）。また、`cache`については1を減じるが（ステップS1203）、`pref`については2減じる（ステップS1205）。この減じ方の差は、キャッシングにより得た情報の方がプリフェッチで得た情報よりも重要でと考えられるためである。

【0078】次に、保持しているデータのうちで最も優先度の低いものを探す`min_pr`処理について説明する。図11にこの`min_pr`処理を示す。各要素は優先度として`cache`と`pref`を持つが、総合優先度として両者の和を用いる。

【0079】この処理では`i`を0からN-1まで検索して（ステップS1301、S1302、S1305）、総合優先度`cache[i] + pref[i]`が最少であるものを探す。検索中で最も小さい総合優先度`v`に格納され、その時の要素の番号が`j`に格納される（ステップS1304）。最後に最少の総合優先度を持つ要素番号`j`が返される（ステップS1306）。

【0080】以上のように本実施形態によれば、連続データの先頭部を保持（プリフェッチ／キャッシング）しておくので、連続データに対する高速な頭出しを実現することができる。しかも、連続データのうち必要な量の先頭部だけ保持するので、必要とする記憶域が少なくすむ。また、アクセス状況等に応じてプリフェッチやキャッシングの要求を制限するので、連続データの処理中にデータの途切れが生じることを回避できる。

【0081】なお、以上に示したのは一実施形態であり、例えば以下のように変形を施すことができる。

①動画データが動画サーバ内で変更になった場合には、古い動画データ番号を欠番にするか、もしくは変更になった動画データに対しては既にキャッシングされているものを無効にする旨のメッセージをクライアントに流す。

【0082】②データをキャッシングする場所は、半導体メモリであっても、磁気ディスクであってもよい。また、これらの両方を併用しても良い。他のいかなる記憶装置であっても、転送速度がデータ処理装置の処理速度よりも速く、かつ、サーバの記憶装置からデータを取ってくるよりもアクセス遅延が小さければ効果がある。しかし、キャッシングに用いる記憶装置のアクセス遅延の大きさによって、記憶しておかなければならない先頭部のデータ量は異なる。例えば、半導体メモリを用いる方が、磁気ディスクを用いるよりもアクセス遅延が小さいので、記憶しておくべき先頭部記憶量も異なる。このように、キャッシングに用いる記憶装置のアクセス待ち時間に応じてデータの先頭部記憶容量を設定することもできる。

【0083】③動画のアクセスの仕方の順序を学習する方法としては、アクセスマトリックスを用いる。すなわち、 i 番目の動画を選んだ次に j 番目の動画を選択する頻度をマトリックスの形で表現する。ユーザからアクセスがあるたびにこのマトリックスを更新し、ある動画を選んだ後に次にどの動画を選ぶ可能性が高いかを、マトリックスをみてクライアントに知らせる。ただし、定数 C を超えないように正規化する。

【0084】このマトリックスを遷移確率行列 A とすると、次に読み出されるであろう動画の優先度は $C * A$ で与えられる。もし、二回先の遷移までを減衰させながら与えたいとすれば、 $C(2/3A + 1/3A * A)$ で与えられる。

【0085】また、クライアント側でローカルに遷移確率情報を管理し、サーバの情報に頼らない方法もある。④先頭部のデータが格納されている領域は adr で最初に決められた値が用いられる。このためデータの先頭部が格納されている記憶装置のアクセス遅延時間とその先頭部データの優先度とは関係がない。そこで、優先度が高いものほどアクセス遅延時間が短い記憶装置にデータを格納するようにデータを再配置することも可能であ

る。

【0086】⑤また、音楽用CDのように比較的順番に聞く可能性が高いものについては、現在までに聞いた曲の先頭部分をキャッシュしておくと共に、CDに記録されている上での次の曲の先頭部分をプリフェッチしておくことも考えられる。このようにすれば、気に入った曲を繰り返し聞いたり、興味の無い曲を次に飛ばして聞くときに、高速な頭出しが可能となる。

本発明は、上述した実施の形態に限定されるものではなく、その技術的範囲において種々変形して実施することができる。

【0087】

【発明の効果】本発明によれば、連続データの先頭部を保持（プリフェッチ／キャッシング）しておくので、連続データに対する高速な頭出しを実現することができる。しかも、本発明によれば、連続データを全てではなく必要な量の先頭部だけ保持するので、必要とする記憶域が少なくすむ。

【0088】また、本発明によれば、アクセス状況等に応じてプリフェッチやキャッシングの要求を制限するので、連続データの処理中にデータの途切れが生じることを回避できる。

【図面の簡単な説明】

【図1】本発明の実施の形態に係るシステムの構成を示す図

【図2】クライアント側で持つデータ構造を示す図

【図3】番号 i の動画データを読み出す手順 $read(i)$ を示すフローチャート

【図4】時計割り込み時の処理を示すフローチャート

【図5】ネットワークからのメッセージの到着時の処理を示すフローチャート

【図6】識別子 i の動画データの先頭データが格納されている位置を調べる $find(i)$ を示すフローチャート

【図7】識別子 i の動画データの先頭データを j 番目の位置に格納するための処理 $add(i, j)$ を示すフローチャート

【図8】識別子 i の動画データの先頭データを削除するための処理 $delete(i)$ を示すフローチャート

【図9】新たな動画データの先頭データを格納すべき位置を検索する関数 $find_place$ を示すフローチャート

【図10】各データの優先度を落とす $decrement$ 処理を示すフローチャート

【図11】保持しているデータのうちで最も優先度の低いものを探す min_pr という処理を示すフローチャート

【図12】従来の記憶装置制御方式を説明するための図

【図13】従来の他の記憶装置制御方式を説明するため

の図

【符号の説明】

301…動画サーバ

302…記憶装置

303…通信部

305…クライアント

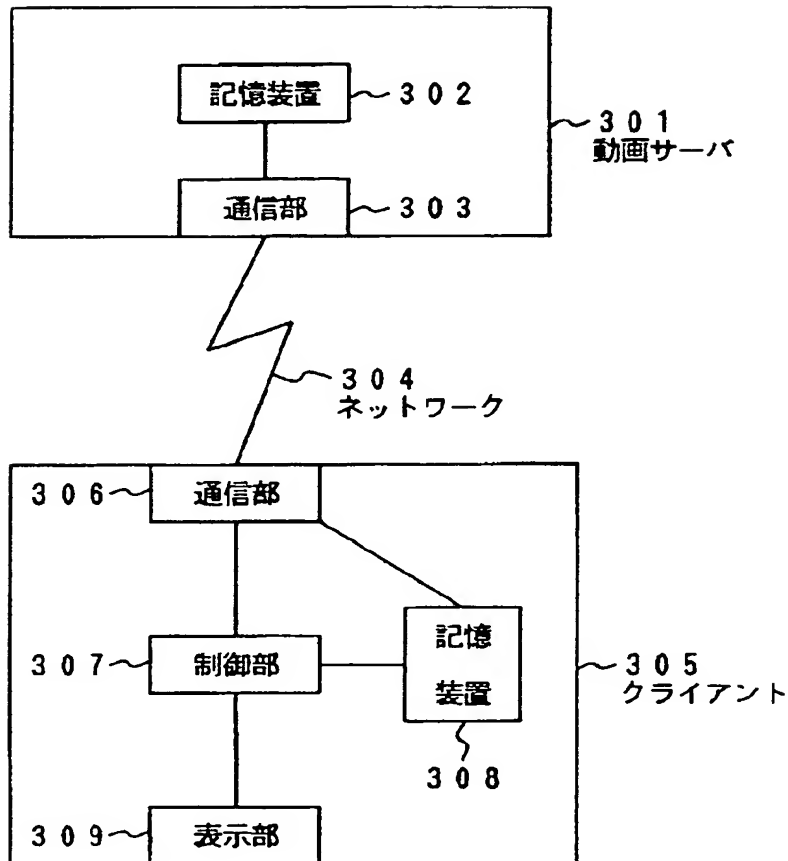
306…通信部

307…制御部

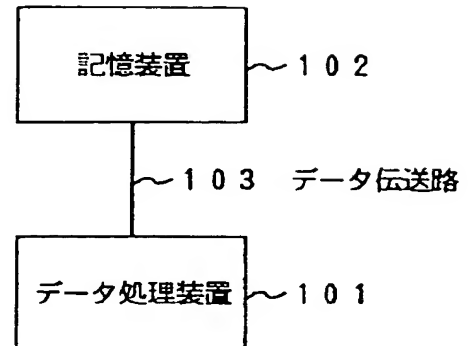
308…高速記憶装置

309…表示部

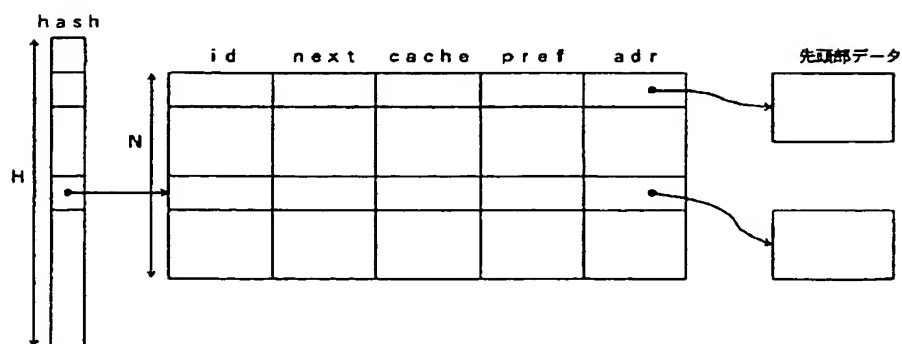
【図1】



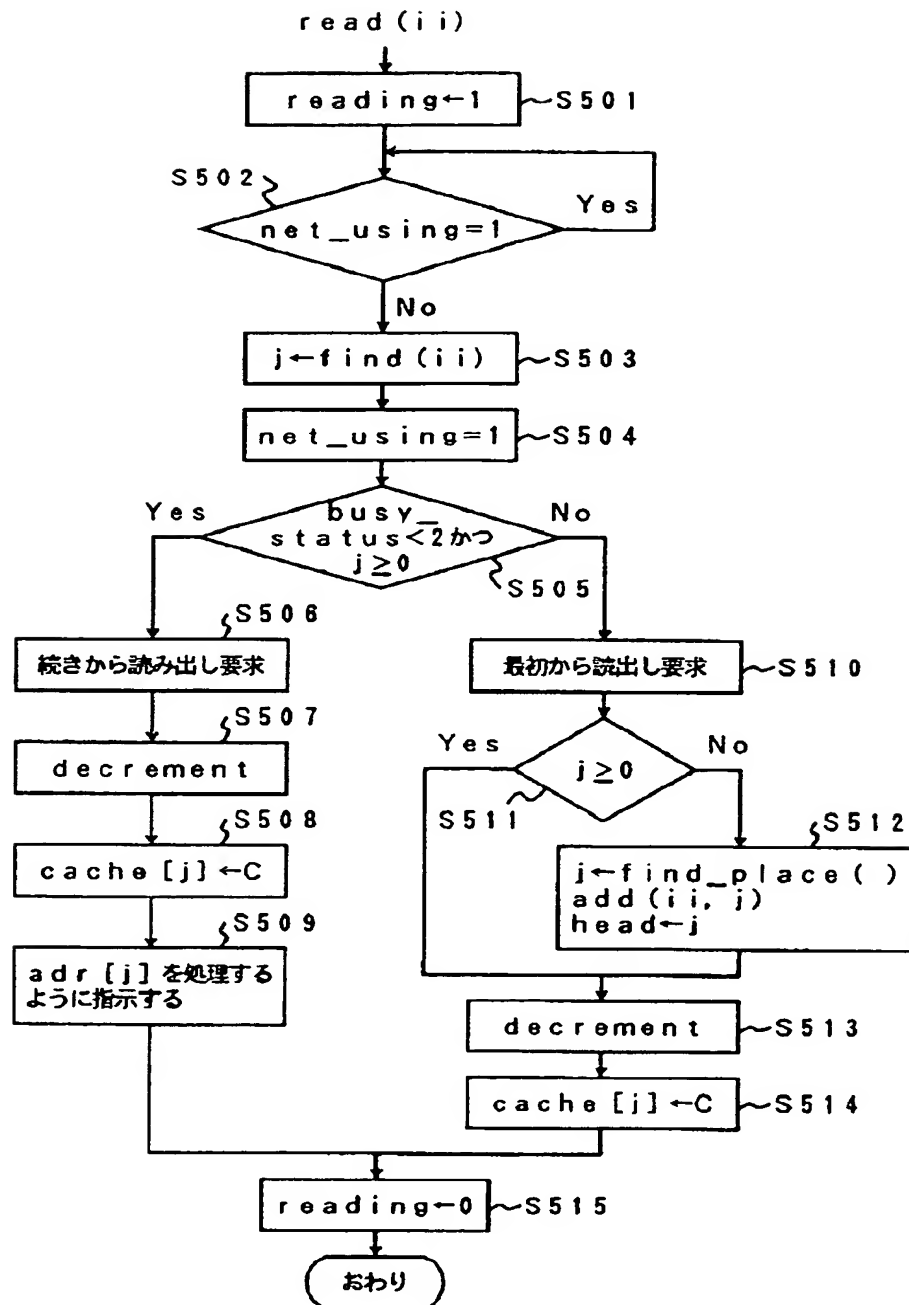
【図12】



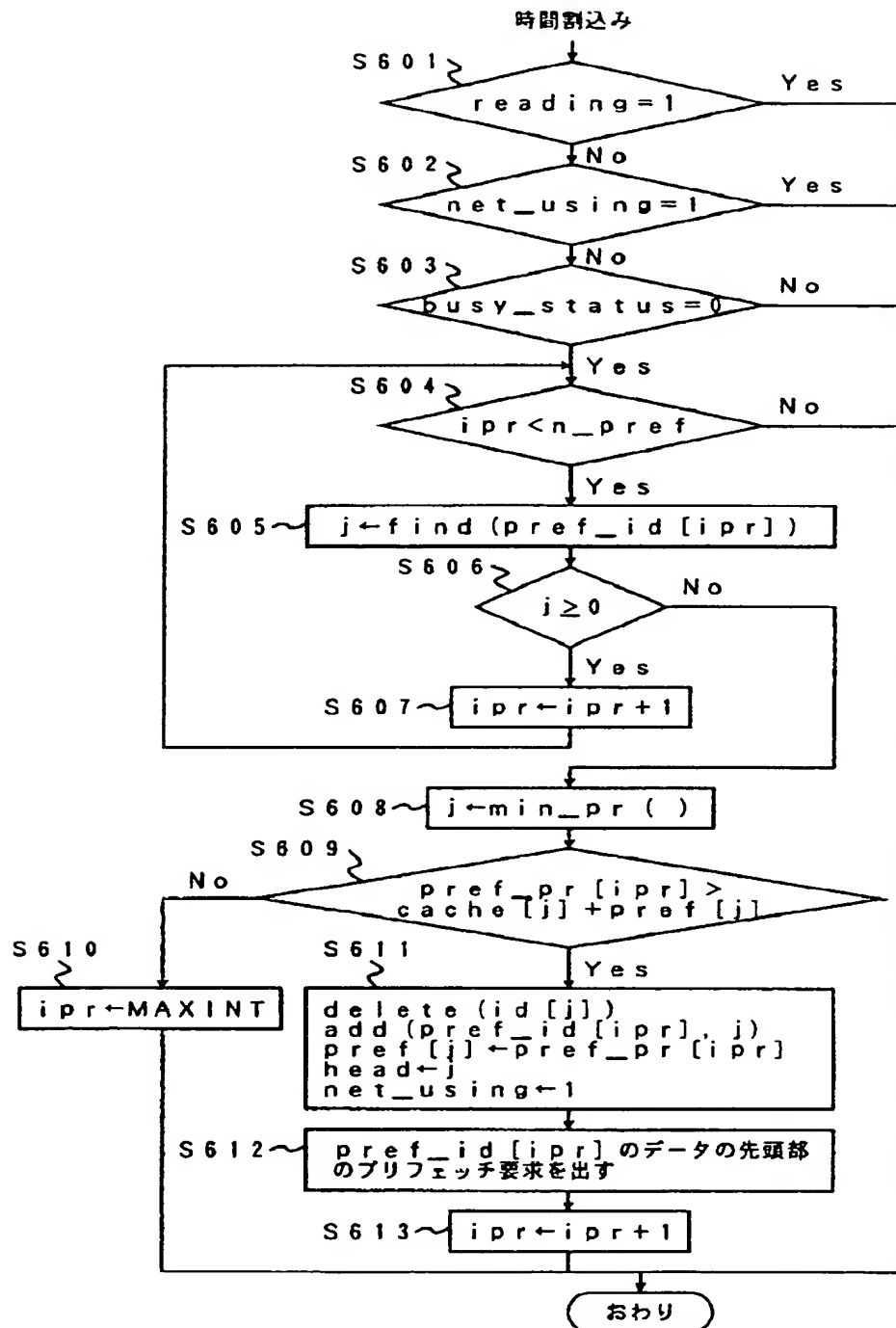
【図2】



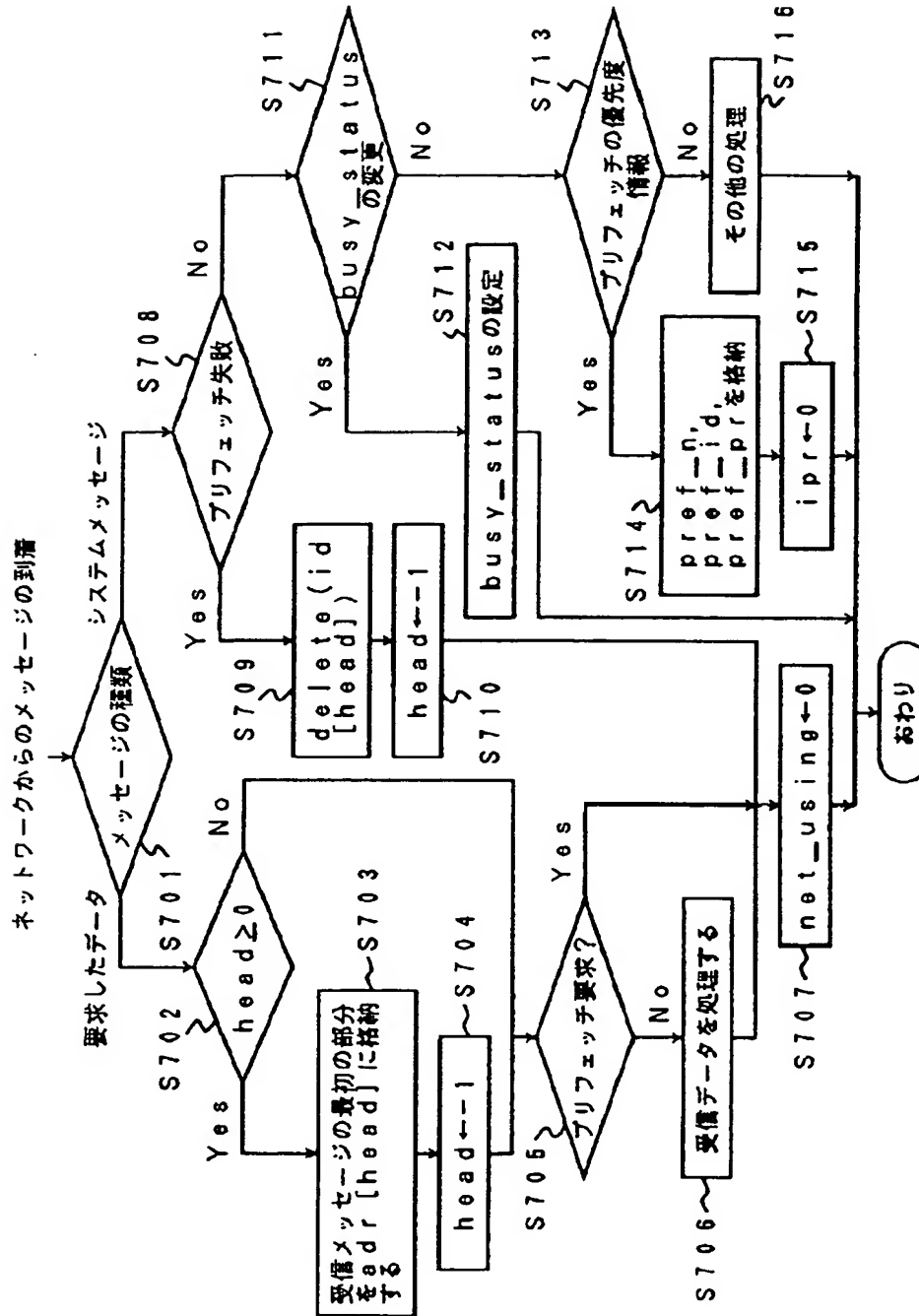
【図3】



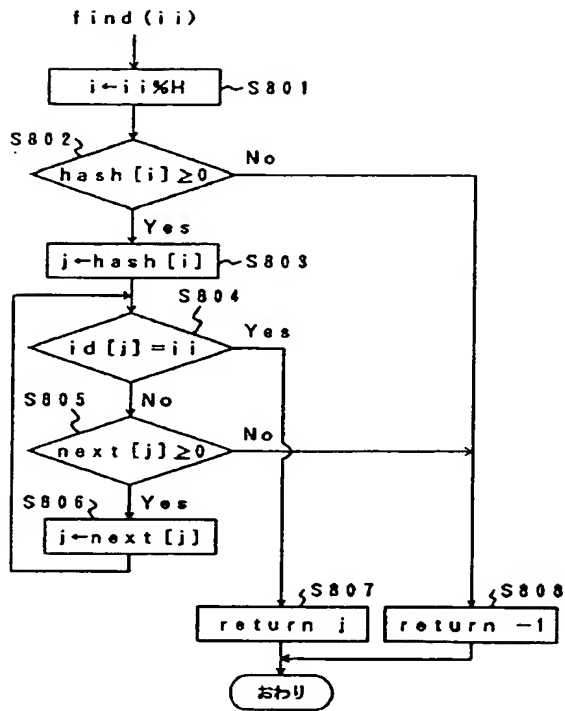
【図4】



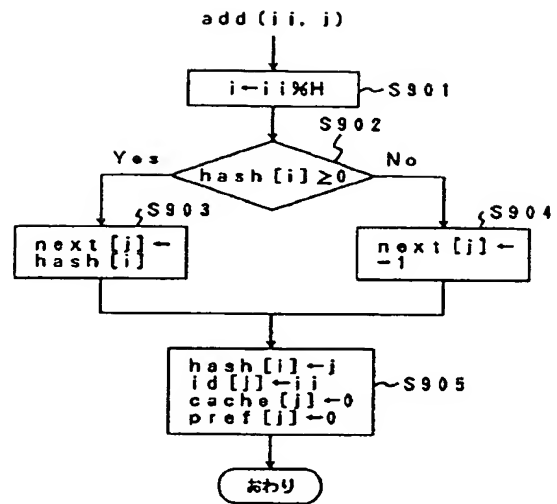
【図5】



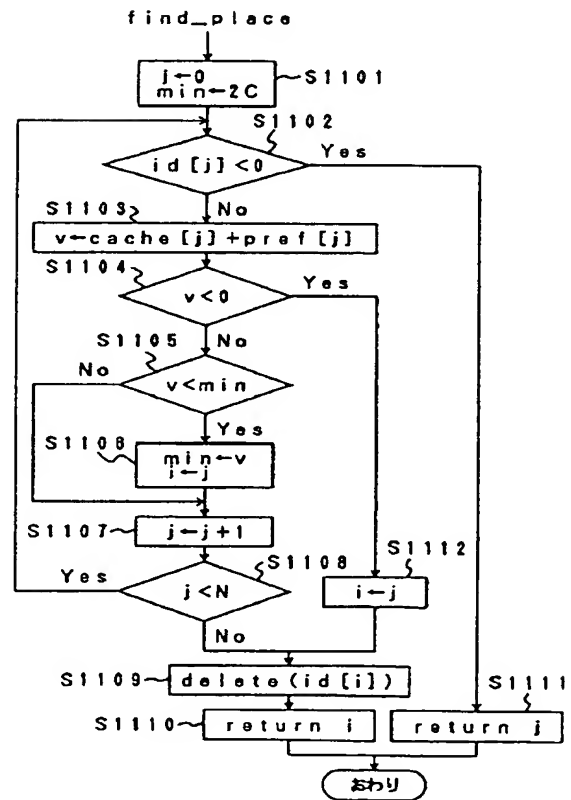
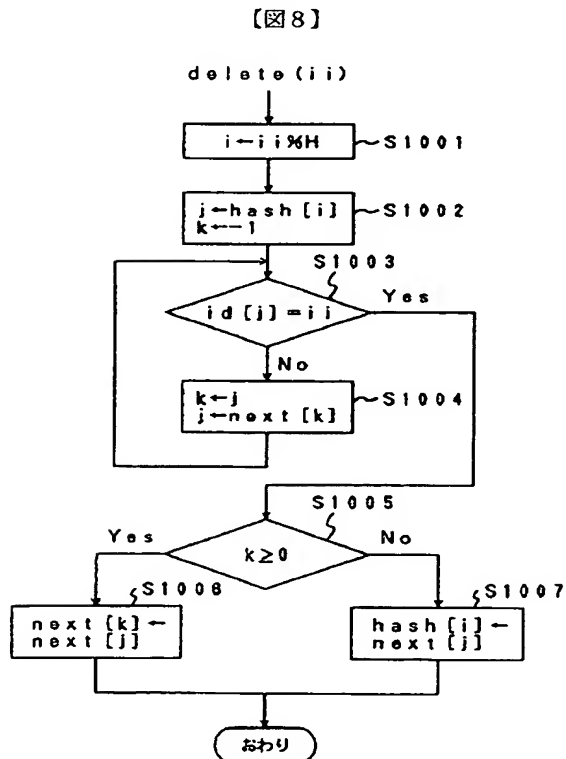
【図6】



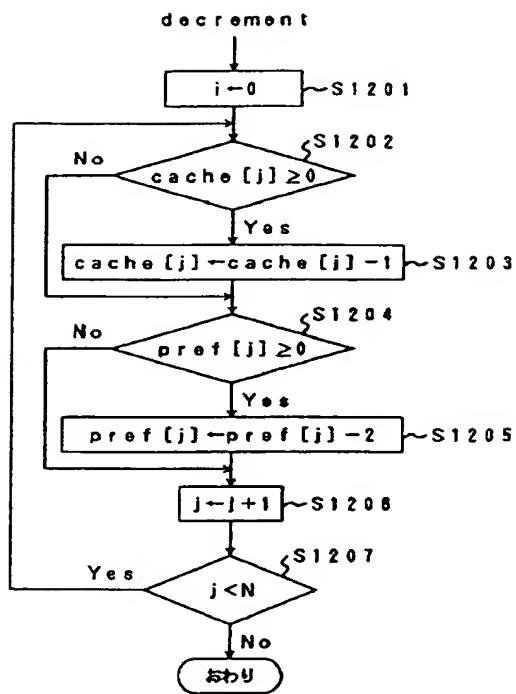
【図7】



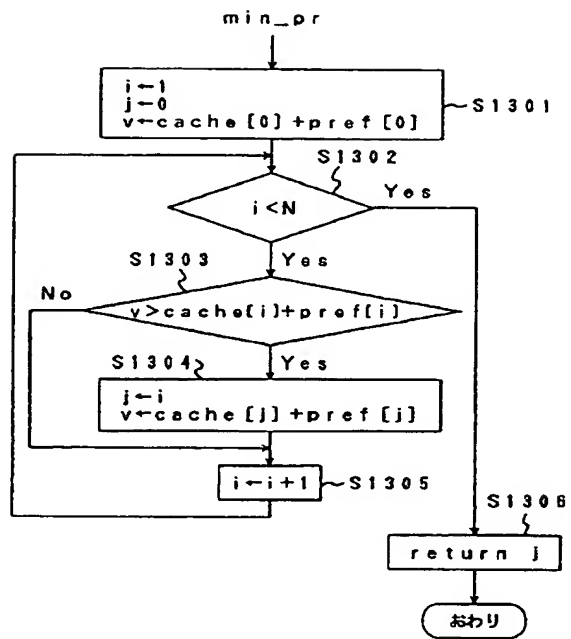
【図9】



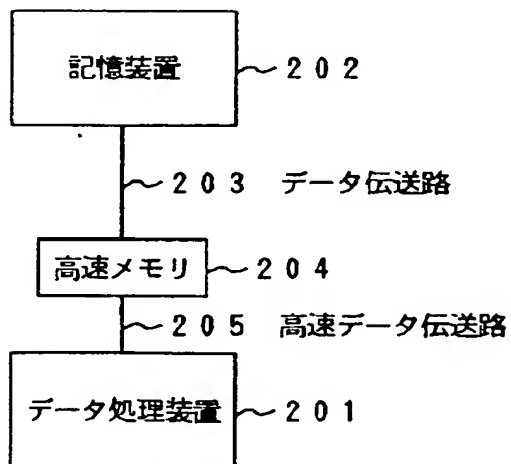
【図10】



【図11】



【図13】



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.